

Compiler Warnings

Try to compile with a decent level of warnings enabled: I used:

```
-Wall -Wextra -pedantic -Wno-aggregate-return
```

Unused Variables

`src/c-util/tools.h` and `src/iofwdutil/tools.hh` defined the `UNUSED()` macro which should be used to annotate unused function parameters and unused local variables.

Unit Tests

We use C-Unit: <http://cunit.sourceforge.net/>

Useful link: http://cunit.sourceforge.net/doc/writing_tests.html#assertions

See the `tests` subdirectory (and the `module.mk.in` file of that directory) to see an example of how to add a unit test.

Namespaces

`using namespace`

The `using namespace` directive is not allowed in header files, to prevent polluting the global namespace. In `cpp` files, the usage is allowed but discouraged.

Manual Resource management

In general, automatic resource management is preferred. For example, instead of manually releasing memory using a `scoped_ptr` or `scoped_array` is safer and clearer. In many cases, using these tools will remove the need to write object destructors.

Forward Declarations

Whenever possible, in header files, use forward declarations instead of bringing in the full definition. (i.e. when a type is only stored or passed as a pointer or reference)

The best way to declare forwards is in a `xxx-fwd.hh` header, which declares the forwards for the relevant names in that logical unit. The other header files can then simply include `xxx-fwd.hh` instead of including the full header.

Example: `sm-fwd.hh` contains forward declarations for (among other things) `sm : SManager`.